Revista Científica da Universidade José Eduardo dos Santos
e-ISSN: 3006-9688 | Vol. 04 | n.º 01 | 2025

PREDIÇÃO DE RISCO ACADÉMICO EM CURSOS DE LICENCIATURAS DE INFORMÁTICA USANDO APRENDIZADO DE MÁQUINA

PREDICTING ACADEMIC RISK IN COMPUTER SCIENCE TEACHER EDUCATION PROGRAMS USING MACHINE LEARNING

Yendrys Blanco Rosabal [1]*; Carlos J. A. Mendes [2]

[1] Instituto Politécnico da Universidade Cuito Cuanavale. Cubango, Angola. [2] Instituto Politécnico da Universidade Cuito Cuanavale. Cubango, Angola

* yendrys24453@gmail.com

RESUMO

Este estudo propõe uma avaliação de modelos de aprendizado de máquina para identificar estudantes em risco de baixo desempenho em cursos de Licenciatura de Informática, utilizando dados coletados em um ambiente virtual de aprendizagem. A pesquisa baseia-se em um conjunto de dados que inclui métricas como frequência de acesso, tempo de permanência, entregas de tarefas, notas e interações em fóruns, classificando os estudantes em níveis de risco ("Alto", "Meio" ou "Baixo"). O objetivo é comparar o desempenho de modelos como KNN, RandomForest e Máquinas de Vetores de Suporte na classificação desses níveis, visando oferecer suporte proativo aos estudantes. A metodologia envolve análise exploratória dos dados, preparação (normalização e divisão em conjuntos de treino e teste), treinamento dos modelos e avaliação por meio de métricas de desempenho. Os resultados destacam as variáveis mais influentes na predição, como notas médias e participação em atividades, e discutem como educadores podem utilizar essas informações para intervenções personalizadas. A proposta enfatiza a importância da análise de dados educacionais para melhorar a retenção e o desempenho acadêmico, oferecendo uma abordagem baseada em evidências para a tomada de decisões pedagógicas.

ABSTRACT

This study proposes an evaluation of machine learning models to identify at-risk students in Computer Science Teacher Education programs, using data collected from a virtual learning environment (VLE). The research analyzes a dataset containing metrics such as login frequency, session duration, assignment submissions, grades, and forum interactions, classifying students into risk levels ("High", "Medium", or "Low"). We compare the performance of KNN, Random Forest, and Support Vector Machine models in classifying these risk levels, aiming to enable proactive student support. The methodology includes exploratory data analysis, data preparation (normalization and train-test splitting), model training, and performance metric evaluation. Results highlight the most influential predictive variables, including average grades and activity participation, while discussing how educators can leverage these insights for personalized interventions. The study emphasizes the importance of educational data analytics for improving academic retention and performance, offering an evidence-based approach to pedagogical decision-making.

Palavras-chave: Aprendizado de máquina, Desempenho académico, Estudantes em risco.

Keywords: Machine learning, Academic performance, At-risk students.

Introdução

Os Ambientes Virtuais de Aprendizagem (AVA) transformaram a educação moderna ao criar espaços digitais onde estudantes e professores podem interagir sem restrições geográficas. À medida que o ensino online ganha popularidade, os AVA tornaram-se essenciais para manter a continuidade educacional e oferecer experiências de aprendizagem flexíveis e personalizadas (Guerrero, 2006).

Esses ambientes não apenas permitem a distribuição de conteúdo e recursos educacionais, mas também promovem a colaboração, a participação ativa e o acompanhamento do progresso académico. Os AVA oferecem ferramentas para gestão de cursos, avaliações, fóruns de discussão, videoconferências e muito mais, facilitando a interação entre estudantes e professores. Além disso, a capacidade de armazenar e analisar dados sobre o uso e a participação nesses ambientes abriu novas oportunidades para a análise educacional, possibilitando a identificação de tendências e padrões que podem informar melhorias pedagógicas (Guerrero, 2006).

Seguindo o abordado por Sabulsky (2019), analisa-se que os ambientes virtuais de aprendizagem (AVA) coletam uma grande quantidade de dados que podem ser utilizados para a análise da aprendizagem, também conhecida como Learning Analytics. Esse tipo de análise busca compreender e melhorar o processo de ensino-aprendizagem e, para isso, é possível obter e analisar diversos tipos de dados, como:

- Atividade do usuário: Informações sobre a interação dos estudantes com a plataforma, como número de logins, tempo de permanência online e frequência de acesso.
- Participação em fóruns e chats: Dados sobre a participação dos estudantes em atividades de discussão, como número de postagens, respostas e envolvimento em chats.
- Uso de recursos educacionais: Informações sobre quais recursos são mais utilizados, quanto tempo é dedicado a eles e quais trajetórias de aprendizagem são seguidas pelos estudantes.

- Desempenho académico: Dados sobre notas em exames, questionários, tarefas e outros trabalhos de avaliação.
- Progresso no curso: Informações sobre o avanço dos estudantes ao longo do curso, como o cumprimento de marcos e a conclusão de módulos.
- Interações com o docente: Dados sobre comunicações com professores, como e-mails ou mensagens internas.
- Resultados de pesquisas: Informações obtidas por meio de pesquisas ou questionários de feedback, que podem ajudar a medir a satisfação dos estudantes e oferecer informações qualitativas sobre suas experiências.

Os AVA, por meio do monitoramento do comportamento dos estudantes, sua participação e seu desempenho académico, permitem que educadores e instituições identifiquem padrões que podem indicar que um estudante está enfrentando dificuldades. A seguir, são explicadas algumas das principais formas pelas quais os AVA contribuem para identificar estudantes em risco:

- Monitoramento do engajamento do estudante: Os AVA permitem rastrear a atividade dos estudantes, como número de logins, tempo de permanência na plataforma e atividades realizadas. Uma queda na participação, como menos acessos ou menos tempo de estudo, pode ser um sinal de que o estudante precisa de apoio.
- Registro de entrega de tarefas: Os AVA registam quando e quantas tarefas são entregues. Se um estudante não entrega tarefas ou o faz de forma inconsistente, isso pode ser um sinal de alerta.
- Análise do desempenho académico: Os AVA fornecem dados sobre notas em exames, testes e tarefas. Ao analisar esses dados, os educadores podem identificar estudantes que estão obtendo notas consistentemente baixas ou que demonstram uma tendência de queda, sugerindo necessidade de intervenção.
- Participação em fóruns e atividades colaborativas: A participação ativa em fóruns, chats e atividades em grupo é um indicador de engajamento. Os AVA permitem visualizar quais estudantes estão participando e em que medida. Uma baixa participação pode ser sinal de desconexão ou desinteresse.
- Interações com os instrutores: Os AVA também registam interações entre estudantes e professores, como mensagens, e-mails e participação em atendimentos virtuais. A ausência

dessas interações pode ser preocupante, indicando que o estudante não está buscando apoio nem participando de atividades de feedback.

A incorporação da mineração de dados na educação virtual permite identificar, por meio de modelos, novas fontes de dados, novas técnicas de representação da informação, extração desses dados e sua apresentação automatizada (Suárez & Díaz Amador, 2009).

Os modelos descritivos, como o clustering (agrupamento) e as regras de associação, ajudam a compreender a estrutura dos dados e a identificar grupos ou padrões interessantes.

Num outro lado, os modelos preditivos, como a classificação e a regressão, permitem prever comportamentos futuros ou classificar novos dados em categorias específicas. Esses modelos são fundamentais para aproveitar ao máximo as informações disponíveis e otimizar os processos de ensino e aprendizagem em ambientes virtuais (Fernández, 2023).

Tais modelos são especialmente úteis na formação virtual para prever o desempenho académico dos estudantes, identificar aqueles que estão em risco de fracassar ou antecipar as necessidades individuais de cada estudante. Esses modelos se baseiam em algoritmos de classificação, regressão e outros, que permitem realizar previsões precisas e tomar medidas pro-activas para melhorar os resultados educacionais (Baker, 2014).

Problema científico

Com o avanço das tecnologias educacionais, os AVAs tornaram-se ferramentas indispensáveis no apoio ao ensino, especialmente em cursos de graduação na área de informática, que exigem constante acompanhamento e flexibilidade. Esses ambientes registam continuamente dados sobre o comportamento dos estudantes, como acessos, participação em fóruns, envio de tarefas, interação com docentes e desempenho em avaliações. Tendo em conta a riqueza desses dados, muitas instituições de ensino superior ainda não os utilizam de forma estratégica para antecipar cenários de risco académico. A ausência de mecanismos automatizados e inteligentes de análise impede a identificação precoce de estudantes com baixa participação ou baixo desempenho, limitando as possibilidades de intervenção pedagógica eficaz e contribuindo para altos índices de retenção e evasão.

Diante desse cenário, surge o seguinte problema científico: Como utilizar, de forma eficiente e automatizada, os dados gerados nos AVA para prever o risco académico de estudantes em cursos de informática?

Para responder a essa questão, parte-se da hipótese de que a aplicação de técnicas de aprendizado de máquina aos dados extraídos dos AVA, como frequência de acesso, tempo de permanência, participação em fóruns, entrega de tarefas e desempenho académico, permite a construção de modelos preditivos capazes de identificar, com antecedência, estudantes em risco de baixo desempenho ou evasão. Essa abordagem possibilita às instituições implementar estratégias pedagógicas mais direcionadas e baseadas em evidências, contribuindo para a melhoria do rendimento e a permanência estudantil.

Objectivo geral

Desenvolver e avaliar modelos de aprendizado de máquina capazes de prever o risco académico em estudantes de licenciatura em informática, utilizando como base, dados de interação em ambientes virtuais de aprendizagem.

Blanco Rosabal, Y.; Mendes, C. J. A. (2025). Predição de risco académico em cursos de licenciaturas de informática usando aprendizado de máquina. RECUJES Edição Especial UCC: Huambo. e-ISSN:3006-9688, Vol. 04, n.º 01, p.474-488.

Material e Métodos

Algoritmos de aprendizado de máquina

k-Nearest Neighbor

Partindo do estudo de Oliveira (2017) se descreve o algoritmo *k-Nearest Neighbor* (KNN) ou k-Vizinhos mais próximos em português, o qual é um método baseado em instâncias. A aprendizagem nos algoritmos baseados em instâncias consiste somente em armazenar os dados de treinamento. Quando uma instância precisa ser testada, o conjunto de treinamento é recuperado da memória para a classificação ser realizada. Uma desvantagem nesses tipos de algoritmos é que o custo computacional deles pode ser muito alto dependendo da quantidade de dados de treinamento. Outro ponto negativo é que todos os atributos são considerados na classificação, em vez de se considerar somente os mais importantes. Dentre os algoritmos baseados em instâncias, o KNN é um dos mais simples e um dos mais conhecidos em reconhecimento de padrões para classificação não paramétrica supervisionada.

Para classificação, o algoritmo descrito na Figura 1, funciona da seguinte maneira: quando uma instância de teste chega, o algoritmo procura os *k* vizinhos mais próximos dele. A classe mais frequente entre os *k* vizinhos será a classe da instância de teste, onde *k* é uma constante definida pelo usuário. A proximidade entre a instância de teste e as outras instâncias é calculada através de uma função de distância. Uma das mais comumente utilizadas é a distância euclidiana.

Random Forest

Tomando como referencia o estudo de Oliveira (2017), se explica que o *Random Forest* (RF), como algoritmo, é uma melhoria da técnica *Bagging* (abreviação de Bootstrap Aggregating) a qual é uma técnica de aprendizado de máquina que tem como objetivo **reduzir a variância** de modelos preditivos. Essa técnica é utilizada principalmente em modelos de Árvores de Decisão. Para isso, o algoritmo seleciona B vezes os exemplos do conjunto de treinamento e treina B árvores com esses exemplos. A predição final é obtida através da média das predições de todas as árvores no caso da regressão ou tomando a classe maioritária no caso da classificação. O *Bagging* tem obtido ótimos resultados combinando centenas e até milhares de árvores.

Figura 1.

Algoritmo KNN para Classificação.

Nota. Tomado de, *"Uma análise de algoritmos de aprendizagem de máquina aplicados em técnicas de localização indoor para diferentes tipos de smartphones"* (p. 12), por L. L. de Oliveira, 2017, Universidade Federal de Pernambuco.

Algorithm 1 k-Nearest Neighbor - Classificação.

Algoritmo de treinamento:
    Para cada exemplo de treino $(x, f(x))$, adicione o exemplo na lista $exemplos\_treino$.
Algoritmo de teste:
    Dada uma instância de teste x_t a ser classificada:
        Sejam $x_1..x_k$ as k instâncias de $exemplos\_treino$ mais próximos de x_t.
        Retorne:
$$\hat{f}(x_t) \leftarrow \underset{v \in V}{argmax} \sum_{i=1}^{k} \delta(v, f(x_i))$$
        onde $\delta(a, b) = 1$ se $a = b$ e $\delta(a, b) = 0$ caso contrário.

Os RFs, de acordo com referenciado pelo autor Filho (2023) se diferem principalmente do *Bagging* em que o algoritmo da Árvore de Decisão é modificado para selecionar a cada divisão de nó somente alguns atributos escolhidos aleatoriamente e não todos. O objetivo disso é evitar que as árvores sejam correlacionados, quer dizer que as árvores do modelo estejam muito parecidas entre si, ou seja, que tomem decisões semelhantes, porque foram construídas com os mesmos atributos ou informações similares. Se um dos atributos é muito forte, então ele provavelmente será escolhido por várias árvores. A Figura 2 mostra o algoritmo em pseudocódigo para o RF.

Máquinas de Vetores de Suporte

As Máquinas de Vetores de Suporte (SVM, Support Vector Machine) vêm recebendo cada vez mais atenção na área de Aprendizagem de Máquina. Ela pode resolver tanto problemas linearmente separáveis como não-linearmente separáveis. Além disso pode ser usada tanto para regressão quanto para classificação. O objetivo da SVM é encontrar um hiperplano que separe o máximo possível dados de classes diferentes. Esse hiperplano é chamado de hiperplano ótimo. A distância entre o hiperplano até a primeira instância de cada classe é chamada de margem. A margem determina o quão bem duas classes podem ser separadas (Oliveira, 2017).

Figura 2.

Algoritmo Random Forest para Classificação e Regressão.

Nota. Tomado de, *"Uma análise de algoritmos de aprendizagem de máquina aplicados em técnicas de localização indoor para diferentes tipos de smartphones"* (p. 12), por L. L. de Oliveira, 2017, Universidade Federal de Pernambuco.

Algorithm 3 *Random Forest* para Classificação e Regressão.

1: **For** $b = 1$ **to** B:
2: (a) Faça uma seleção aleatória dos dados de treinamento.
3: (b) Crie uma árvore de decisão T_b com os dados selecionados, recursivamente repetindo os seguintes passos para cada nó terminal da árvore, até o nó de tamanho mínimo ser encontrado.
4: (i) Selecione m atributos aleatóriamente dos p atributos da base de treinamento.
5: (ii) Selecione o melhor atributo dentre os m.
6: (iii) Divida o nó em dois nós filhos.
7: A saída será o conjunto de árvores $\{T_b\}_1^B$.
8: Para fazer a predição de um novo ponto x:
9: Para Regressão: $\hat{f}(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$.
10: Para Classificação: Seja $\hat{C}_b(x)$ a classe predita da *bth* árvore da *Radom Forest*. Então $\hat{C}^B(x) =$ votos majoritários $\{\hat{C}_b(x)\}_1^B$.

Hiperparâmetros

Hiperparâmetros são parâmetros que não são diretamente aprendidos pelos algoritmos de aprendizado de máquina durante o treinamento do modelo. Eles são definidos antes do treinamento e afetam diretamente o desempenho e o resultado final do modelo. Em outras palavras, os hiperparâmetros são configurações específicas que controlam o comportamento do algoritmo de aprendizado de máquina. Os hiperparâmetros são distintos dos parâmetros do modelo, que são os valores aprendidos pelo algoritmo durante o processo de treinamento.

Enquanto os parâmetros do modelo são ajustados internamente com dados durante o treinamento, os hiperparâmetros são definidos externamente pelo cientista de dados e devem ser otimizados para obter o melhor desempenho do modelo (Awari, 2023).

Do apresentado se entende que os hiperparâmetros pousem um papel significativo no desempenho e na precisão dos modelos de aprendizado de máquina. É importante explorar e experimentar diferentes métodos de ajuste de hiperparâmetros para encontrar a combinação mais adequada para o problema em questão. Além disso, a otimização dos hiperparâmetros deve ser considerada como uma tarefa contínua e iterativa.

Ferramentas

Para desenvolver a proposta serão utilizadas uma variedade de ferramentas computacionais e bibliotecas de software. A seguir são descritas as mesmas no contexto do estudo:

Linguagem de Programação

Python foi a linguagem de programação principal utilizada para desenvolver e executar o código do estudo. Sua popularidade em ciência de dados e aprendizado de máquina, juntamente

com sua vasta coleção de bibliotecas especializadas, a tornam uma escolha ideal para este tipo de projeto. Bibliotecas Python:

- **Pandas:** Utilizada para manipulação e análise de dados. Fornece estruturas de dados flexíveis e eficientes, como DataFrames, para armazenar e processar os dados do estudo.
- **NumPy:** Essencial para computação numérica em Python, fornecendo suporte para arrays multidimensionais e funções matemáticas avançadas.
- **Matplotlib e Seaborn:** Bibliotecas para visualização de dados, permitindo a criação de gráficos e visualizações informativas para explorar os dados e comunicar os resultados do estudo.
- **Scikit-learn:** Biblioteca fundamental para aprendizado de máquina em Python, fornecendo uma ampla gama de algoritmos de classificação, regressão, clustering, pré-processamento de dados e métricas de avaliação. Foi utilizada para treinar e avaliar os modelos KNN, Random Forest e SVM.
- **Imbalanced-learn:** Biblioteca específica para lidar com conjuntos de dados desbalanceados, como o utilizado no estudo. Fornece técnicas de reamostragem, como SMOTE, para balancear as classes e melhorar o desempenho dos modelos.

Ambiente de Desenvolvimento

Google Colab: Plataforma de nuvem para desenvolvimento e execução de código Python, baseada em Jupyter Notebooks. Oferece recursos computacionais gratuitos e facilita a colaboração e o compartilhamento de projetos. Foi o ambiente escolhido para desenvolver e executar o código do estudo.

Ferramentas Adicionais

- **GridSearchCV:** Ferramenta do scikit-learn para otimização de hiperparâmetros dos modelos, utilizando validação cruzada. Foi utilizada para encontrar os melhores valores para os parâmetros dos modelos KNN, Random Forest e SVM, maximizando seu desempenho.
- Métricas de Avaliação: Funções do scikit-learn para calcular métricas de desempenho dos modelos, como acurácia, matriz de confusão e relatório de classificação. Foram utilizadas para avaliar e comparar o desempenho dos modelos treinados.

Conjunto de Dados

Utilizou-se o conjunto de dados (dataset) *student_learning_analytics.csv*, como fonte de dados. Variáveis comportamentais:

- Logins semanais (Logins_Per_Week);
- Tempo médio por sessão (Time_Spent_Per_Login);

- Participação em fóruns (Forum_Participation);
- Interações com docentes (Instructor_Interactions);

Variáveis de desempenho acadêmico:

- Tarefas entregues (Assignments_Submitted);
- Notas médias (Average_Assignment_Grade);
- Resultados em testes (Test_1_Score, Test_2_Score, Test_3_Score);
- Exame final (Final_Exam_Score);

Variável alvo:

- Classificação de risco (Risk_Level: Alto, Médio, Baixo). Este estudo propõe uma abordagem baseada em aprendizado de máquina para identificar proativamente estudantes em risco acadêmico em cursos de licenciatura em informática, utilizando dados de Ambientes Virtuais de Aprendizagem (AVAs). A metodologia foi estruturada nas seguintes etapas:

  Carga de Dados

- Objetivo: Carregar e realizar uma primeira inspeção nos dados.
- Passos:
- Importação das bibliotecas: Importar as bibliotecas necessárias, como pandas, numpy, matplotlib, seaborn, e as ferramentas de aprendizado de máquina como sklearn e imblearn.
- Carregamento dos dados: O arquivo CSV contendo o dataset é carregado e visualizado. O comando pd.read_csv() é utilizado para ler o arquivo.
- Visualização preliminar:
- Exibir as primeiras linhas do conjunto de dados usando data.head().
- Verificar a estrutura dos dados e tipos de variáveis com data.info().
- Obter estatísticas descritivas com data.describe() para entender a distribuição das variáveis numéricas.
- Verificar a presença de valores ausentes usando data.isnull().sum().

Análise Exploratória de Dados

- Objetivo: Explorar as características dos dados e identificar potenciais desequilíbrios .
- Passos:
- Distribuição das classes: Analisar a distribuição da variável alvo (Risk_Level) por meio de gráficos de barras com (sns.countplot()).
- Codificação da variável alvo: Se necessário, converter variáveis categóricas em variáveis numéricas usando o LabelEncoder() para transformar a coluna Risk_Level.
- Balanceamento das classes: Verificar o balanceamento entre as classes. Caso haja um desbalanceamento, aplicar técnicas como o SMOTE (Synthetic Minority Over-sampling Technique) para balancear o conjunto de dados.

Preparação dos Dados

- Objetivo: Preparar os dados para o treinamento, incluindo a normalização e divisão entre treino e teste.

Passos:

- Separação de variáveis independentes e dependentes: A variável dependente (Risk_Level) é separada da variável independente (X), que contém as características dos dados.
- Aplicação de SMOTE: Se necessário, aplicar SMOTE para balancear as classes e evitar que o modelo seja enviesado por classes desproporcionais.
- Divisão em conjuntos de treino e teste: Utilizar a função train_test_split() para dividir os dados balanceados em conjuntos de treino (70%) e teste (30%).
- Normalização dos dados: Usar StandardScaler() para normalizar os dados, garantindo que as variáveis com diferentes escalas não impactem o modelo.

Treinamento dos Modelos

- Objetivo: Construir e treinar modelos de aprendizado supervisionado.

Passos:

- Modelos a serem treinados:

Blanco Rosabal, Y.; Mendes, C. J. A. (2025). Predição de risco académico em cursos de licenciaturas de informática usando aprendizado de máquina. RECUJES Edição Especial UCC: Huambo. e-ISSN:3006-9688, Vol. 04, n.º 01, p.474-488.

- K-Nearest Neighbors (KNN): Um modelo baseado em vizinhos próximos.
- Random Forest: Um modelo baseado em árvores de decisão.
- Support Vector Machine (SVM): Um modelo de classificação baseado na maximização da margem entre as classes.
- Ajuste de hiperparâmetros: Utilizar o GridSearchCV para testar combinações de hiperparâmetros e encontrar os melhores para cada modelo.
- Treinamento dos modelos: Após o ajuste, treinar cada modelo com o conjunto de dados de treino.

Avaliação de Modelos

- Objetivo: Avaliar o desempenho de cada modelo de forma objetiva.

Passos:

- Avaliação das predições:
- Para cada modelo (Random Forest, KNN e SVM), calcular a acurácia, a matriz de confusão e o relatório de classificação utilizando as métricas de precisão, recall, F1-score principalmente.
- Exibição de resultados:
- Apresentar os resultados de cada modelo e suas métricas de desempenho.
- Comparar as acurácias dos modelos para determinar qual deles tem o melhor desempenho para o conjunto de dados.

Visualização de Resultados

- Objetivo: Visualizar a comparação entre os modelos.

Passos:

- Gráfico de Precisão: Criar um gráfico de barras usando matplotlib para comparar a precisão dos modelos de forma visual.
- Interpretação dos resultados: Analisar e interpretar os gráficos para identificar o modelo que tem o melhor desempenho.

Resultados e Discussão

A avaliação dos modelos de classificação, K-Nearest Neighbors (KNN), Random Forest (RF) e Support Vector Machine (SVM), com foco na precisão, revelou perceções importantes sobre a capacidade preditiva do nível de risco, utilizando os dados do estudo.

- Random Forest: Demonstrou a maior precisão entre os modelos testados, atingindo 0.9231. Este resultado destaca a eficácia do RF na predição do nível de risco, com alta capacidade de generalização e robustez a sobreajuste. Sua arquitetura, baseada em múltiplas árvores de decisão, permite capturar interações complexas entre as variáveis preditoras, contribuindo para a alta precisão alcançada.
- KNN: Obteve uma precisão de 0.8846, demonstrando bom desempenho na tarefa de classificação, porém inferior ao RF. Sua simplicidade e interpretabilidade são vantagens, mas a sensibilidade à escala das variáveis e a definição do número ideal de vizinhos (k) podem influenciar sua performance. No estudo, o valor de *k* foi definido por padrão, o que pode ter limitado a precisão do modelo.
- SVM: Atingiu uma precisão de 0.8462, sendo o modelo com menor precisão entre os três avaliados. Embora seja capaz de lidar com dados de alta dimensionalidade e encontrar hiperplanos ótimos para separação das classes, a escolha da função *kernel* e a sensibilidade a valores atípicos (outliers) podem ter impactado negativamente sua performance neste conjunto de dados.

Comparação entre os Modelos

A comparação direta da precisão dos modelos evidencia a superioridade do Random Forest (0.9231) na predição do nível de risco. KNN (0.8846) e SVM (0.8462) apresentaram precisões inferiores, indicando que, para este problema específico e com os dados utilizados, o Random Forest é a escolha mais adequada para alcançar maior aprovação nas previsões.

A precisão superior do Random Forest pode ser atribuída à sua capacidade de lidar com variáveis preditoras complexas e interativas, além de sua robustez a ruídos e sobreajuste. Em contraste, KNN e SVM podem ser mais sensíveis a características específicas dos dados, como escala e presença de outliers. Adicionalmente, a escolha de hiperparâmetros, como o número de vizinhos (k) no KNN e a função kernel no SVM, pode influenciar significativamente a precisão dos modelos.

Blanco Rosabal, Y.; Mendes, C. J. A. (2025). Predição de risco académico em cursos de licenciaturas de informática usando aprendizado de máquina. RECUJES Edição Especial UCC: Huambo. e-ISSN:3006-9688, Vol. 04, n.º 01, p.474-488.

Conclusões

A comparação entre os modelos KNN, Random Forest e SVM demonstrou que todos têm potencial para classificar adequadamente os estudantes de acordo com seus níveis de risco. Contudo, o desempenho dos modelos pode variar dependendo das características do conjunto de dados e da configuração dos hiperparâmetros.

Random Forest foi destacado pela capacidade de capturar interações complexas entre as variáveis, além de fornecer uma interpretação clara sobre a importância das características.

KNN pode ser mais sensível a grandes volumes de dados e à escolha da distância entre pontos.

SVM pode ter bom desempenho, especialmente em conjuntos de dados com margens de separação bem definidas entre as classes, mas requer uma boa escolha do kernel e da regularização.

As variáveis mais influentes na predição dos níveis de risco dos estudantes incluem notas médias e participação nas atividades (como entregas de tarefas e interações nos fóruns). Estes fatores têm um impacto direto na avaliação do risco de baixo desempenho. A análise de dados educacionais, com a aplicação de aprendizado de máquina, permite uma abordagem mais personalizada e proativa para a identificação de estudantes em risco. A capacidade de prever o risco de baixo desempenho oferece aos educadores uma ferramenta para intervir de forma mais eficaz e no momento certo.

O uso de modelos de aprendizado de máquina para prever o risco de baixo desempenho dos estudantes representa uma grande oportunidade para as instituições educacionais oferecerem apoio direcionado e pro-activo. Isso pode não apenas melhorar a retenção acadêmica, mas também fornecer uma abordagem mais humana e personalizada para o acompanhamento dos alunos.

Referências Bibliográficas

Awari. (2023). Hiperparâmetros em aprendizado de máquina. https://awari.com.br/machine-learning-hyperparameter-hiperparametros-em-aprendizado-de-maquina-2/

Baker, R. (2014). Educational data mining and learning analytics. University of Pennsylvania. https://www.researchgate.net/publication/316628053_Educational_data_mining_and_learning_analytics

Fernández, A. F. (2023). Métodos y modelos para la predicción electoral: Una guía práctica. OBETS – Ciencia Abierta.

Filho, Luiz Henrique Barbosa (2023). Bagging, Random Forests e Boosting. https://analisemacro.com.br/economia/macroeconometria/bagging-random-forests-e-boosting/

Guerrero, C. S. (2006). Los entornos virtuales de aprendizaje como instrumento de mediación. Investigación Educativa, 10(18), 41–56. https://revistasinvestigacion.unmsm.edu.pe/index.php/educa/article/download/3776/3038/12802

Oliveira, L. L. de. (2017). Uma análise de algoritmos de aprendizagem de máquina aplicados em técnicas de localização indoor para diferentes tipos de smartphones [Trabalho de conclusão de curso, Universidade Federal de Pernambuco]. https://www.cin.ufpe.br/~tg/2017-1/llo_tg.pdf

Sabulsky, G. (2019). Analíticas de aprendizaje para mejorar el aprendizaje y la comunicación a través de entornos virtuales. Revista Iberoamericana de Educación, 80(1), 13–30. https://doi.org/10.35362/rie8013340

Suárez, Y. R., & Díaz Amador, N. (2009). Herramientas de minería de datos. https://www.redalyc.org/pdf/3783/378343637009.pdf